



# New EMMS Usage Guidelines

Information for Developers and ISVs

From Intel® Developer Services  
[www.intel.com/IDS](http://www.intel.com/IDS)

March 1996

*Information in this document is provided in connection with Intel® products. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document. Except as provided in Intel's Terms and Conditions of Sale for such products, Intel assumes no liability whatsoever, and Intel disclaims any express or implied warranty, relating to sale and/or use of Intel products including liability or warranties relating to fitness for a particular purpose, merchantability, or infringement of any patent, copyright or other intellectual property right. Intel products are not intended for use in medical, life saving, or life sustaining applications. Intel may make changes to specifications and product descriptions at any time, without notice.*

*Note: Please be aware that the software programming article below is posted as a public service and may no longer be supported by Intel.*

Copyright © Intel Corporation 2004

\* Other names and brands may be claimed as the property of others.

## **New EMMS Guidelines**

---

March 1996

### **CONTENTS**

Previous EMMS Guidelines

Performance Side Effects

New EMMS Guidelines

Guidelines for EMMS usage have been changed due to significant side effect delay on Pentium® II processor.

The Intel Architecture MMX™ Technology, Programmer's Reference Manual describes the EMMS instruction as follows,

"The EMMS instruction set the values of the floating-point (FP) tag word to empty (all ones). EMMS marks the registers as available, so they can subsequently be used by floating-point instructions.

If a floating-point instruction loads into one of the registers before it has been reset by the EMMS instruction, a floating-point stack overflow can occur, which results in an FP exception or incorrect result."

## Previous EMMS Guidelines

The previous guidelines for the EMMS instruction The Intel Architecture MMX™ Technology, Developer's Manual March 1996, section 4.4, EMMS Guidelines states,

...: Always call the EMMS instruction at the end of your MMX™ code.

Since the MMX™ registers are aliased on the floating-point registers, it is very important to clear the MMX™ register before issuing a floating-point instruction. Use the EMMS instruction to clear the MMX™ register and set the value of the floating-point tag word (TW) to empty (that is, all ones). This instruction should be inserted at the end of all MMX™ code segments to avoid an overflow exception in the floating-point stack when a floating-point instruction is executed.

## Performance Side Effects

According to a recent study significant performance delay will occur on Pentium® II processor as a side effect of the EMMS instruction. Regardless of whether a FP instruction is executed. The side effect delay is incurred with the first MMX™ instruction encountered after the EMMS instruction.

## New EMMS Guidelines

The following guidelines are suggested for the use of the EMMS instruction,

1. Use EMMS instruction after MMX instruction if you don't know what kind of instruction will follow (FP or MMX). Example, a calling to library, DLL, or different execution path option.
2. Do not use the EMMS instruction if the next code to be executed are MMX™ instructions as opposed to floating-point instructions
3. Avoid placing EMMS in the body of a loop containing MMX™ instructions
4. On a non-preemptive OS such as DOS and Win 3.1, use EMMS before a task switch may occur. (consider implicit and explicit task switching).
5. On a preemptive OS such as Win 95 and Win NT, rules 1-3 are sufficient because the OS saves the FP/MMX state between thread switches
6. Use the Leak checker utility to find omissions of EMMS instruction.